REMARKS

I.   Introduction

Claims 1-32 are pending in the present application.   In a September 6, 2006, Office Action (hereinafter "Office Action"), Claim 22 was rejected under 35 U.S.C. § 112 as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicants regard as the invention.   Claim 22 was also rejected under 35 U.S.C. § 101 as being directed to non-statutory subject matter.   Claims 1-31 were rejected under 35 U.S.C. § 102(e) as being unpatentable over U.S. Publication No. 2005/0060304, to Parikh (hereinafter "Parikh").

II.   35 U.S.C. § 112 and § 101 Rejections

Claim 22 was rejected under 35 U.S.C. § 112 as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicants regard as the invention. Additionally, Claim 22 was rejected under 35 U.S.C. § 101 as being directed to non-statutory subject matter.   Claim 22 has been cancelled without prejudice to facilitate the prosecution of the present application.

III.   Claim Rejections

A.   Introduction

Claims 1-31 were rejected under 35 U.S.C. § 102(e) as being unpatentable over Parikh. For the following reasons, applicants respectfully submit that the rejected claims of the present application are not anticipated by Parikh because the cited reference fails to teach or suggest generating an inverted keyword index and an inverted keyword attribute index corresponding to keyword data. Further, the cited reference fails to teach or suggest a shared process memory containing both an inverted keyword index and an inverted keyword attribute index.   Still further, Parikh fails to teach or suggest the use of adjusting memory pointers corresponding to the inverted keyword index. Even further, Parikh fails to teach or suggest creating a temporary

inverted attribute index and converting the temporary index into a inverted keyword attribute index. Prior to discussing more detailed reasons why applicants believe that all of the claims of the present application are allowable over the cited references, a brief description of the present invention and the cited references is presented.

1.   Summary of the Present Invention

The present application is generally related to a system and method for facilitating full text searching of data. More particularly, the present invention relates to a query engine including a data parsing/index generation component. The data parsing/index generation component can generate an inverted keyword index and an inverted keyword attribute index and store both indices in a shared process memory. The present invention also relates to a query processing component in the query engine for processing data queries from the indices in the shared memory buffer.

In accordance with one aspect, the data parsing/index generation component obtains a first token from a set of data. The data parsing/index generation component inserts the keyword token data as a node in a red and black index tree. Once the red and black index tree is adjusted, the data parsing/index generation component inserts keyword token attribute data for each token into a temporary inverted keyword attribute index, which is set up as a red and black tree. Each node within the temporary inverted keyword attribute tree index includes a linked list of occurrences of the keyword within a particular document. The data parsing/index generation component converts the temporary inverted keyword attribute tree index into an optimized list. After the temporary red and black index tree is converted, the data parsing/index generation component adjusts shared memory pointers for the inverted keyword index and the inverted keyword attribute index.

The present invention provides the ability to more effectively facilitate full text searching. Because the inverted keyword index and the inverted keyword attribute index are within a shared process memory, no data needs to be passed between the data parsing/index generating component and the query engine, which results in an increased burden on system memory resources. Furthermore, the traditional delay between the data parsing/index generating component and the query engine is no longer present.

2.   U.S. Publication No. 2005/0060304, to Parikh

Parikh is purportedly directed toward a method for searching through an arrangement of nodes represented in a tree. Parikh, para. 0011, 0029. The method involves receiving a word and searching the inverted index to determine whether the word is a keyword. If the word is a keyword, Parikh teaches "jumping" to a node identified in the inverted index as correlated to that keyword. Abstract. As taught by Parikh, the unique identification of each node in the tree allows the creation of a list of all the keywords and their associated nodes so that, if a keyword is duplicated in two or more nodes, it need only be listed once. Para. 0033. An inverted index is created that correlates between keywords and nodes such that at least some nodes containing a given keyword are indexed to that given keyword. Para. 0011, 0034-0036, 0043-0046. Further, the relevant keywords and node identifiers may include a frequency of occurrence for the keyword within the description with the keyword in the inverted index. Para. 250. The inverted keyword index is stored in a file. Para. 0043-0046. Parikh teaches that a user may "jump" to a node identified in the keyword index as correlated to that keyword without traversing through the tree. Abstract; para. 0011, 0047.

Parikh fails to teach or suggest generating an inverted keyword index and an inverted keyword attribute index corresponding to keyword data. Further, the cited reference fails to teach or suggest a shared process memory shared process memory containing both an inverted

keyword index and an inverted keyword attribute index. Moreover, Parikh fails to teach or suggest the use of adjusting memory pointers corresponding to the inverted keyword index. Still further, Parikh fails to teach or suggest creating a temporary inverted attribute index.

B.   The Claims Distinguished

1.   35 U.S.C. § 102(e) Rejections

a.   Independent Claims 1, 11, and 23

For purposes of this discussion, Claims 1, 11, and 23 will be discussed together because the limitations discussed herein are similar for each claim. Claim 1, as amended, reads as follows:

> 1.   A method for facilitating full text searching of a set of data, the method comprising:
>
> obtaining keyword data corresponding to a set of data;
>
> generating an inverted keyword index and an inverted keyword attribute index corresponding to the keyword data;
>
> storing the inverted keyword index and the inverted keyword attribute index in a shared process memory;
>
> obtaining a keyword query from a first process; and
>
> processing the keyword query from the inverted keyword index in a shared memory.

Similarly, Claim 11 reads as follows:

> 11.   A method for facilitating full text searching of a set of data, the method comprising:
>
> obtaining keyword data corresponding to a set of data;
>
> generating an inverted keyword index and an inverted keyword attribute index corresponding to the keyword data; and

storing the inverted keyword index and the inverted keyword attribute index in a shared process memory buffer.

Claim 23, as amended, reads as follows:

23. A system for facilitating full text searching, the system comprising:

one or more processes for issuing keyword queries;

an index generation component for obtaining a set of data and generating an inverted keyword index and an inverted keyword attribute index;

a shared memory buffer for storing the inverted keyword index and the inverted keyword attribute index of a set of data; and

a query processing component for processing keyword queries issued by the one or more processes from the inverted keyword index stored in the shared memory buffer.

In contrast to the claims of the present application, Parikh is directed toward a method for jumping to a particular node within a tree corresponding to an inverted index. Parikh, para. 37, 0043-0046. The inverted index is created by relevant keywords and node identifiers. Para. 0043, 0052-0053. Further, the relevant keywords and node identifiers may include a frequency of occurrence for the keyword within the description with the keyword in the inverted index. Para. 250. Nevertheless, Parikh does not teach or suggest generating an inverted keyword index and an inverted keyword attribute index corresponding to the keyword data. Instead, Parikh only teaches an inverted keyword index which may contain an occurrence attribute associated with the entries in the inverted keyword index. The occurrence attribute associated with the entries is not an inverted keyword attribute index because indexed information about the occurrence attribute is not given. Rather, the occurrence attribute only discloses how many times the keyword may be found in the node. Thus, Parikh fails to teach or suggest generating an inverted

-12-

keyword index and an inverted keyword attribute index corresponding to the keyword data as recited in Claims 1, 11, and 23.

As taught in Parikh, the relevant keywords and node identifiers which may include a frequency of occurrence for the keyword within the description with the keyword in the inverted index can be stored in a file. Para. 0043-0046. Parikh does not teach or suggest a shared process memory containing both an inverted keyword index and an inverted keyword attribute index as recited in Claims 1, 11, and 23. Para. 0043-0046. Instead, Parikh only stores an inverted keyword index associated with the location of those keywords to a tree in a file. The file is not a shared process memory because the file contains only the inverted keyword index, not an inverted keyword attribute index. Additionally, the occurrence attribute is not an inverted keyword attribute index as described above. Thus, Parikh fails to teach or suggest storing a shared process memory containing both an inverted keyword index with an inverted keyword attribute index as recited in Claims 1, 11, and 23.

To anticipate a claim under § 102(e), the cited reference must teach each and every element recited in the claim. *Verdegaal Bros. v Union Oil Co. of California*, 814 F.2d 628, 631 (Fed. Cir. 1987). With regard to Claim 1, 11, and 23, applicants respectfully submits that the cited reference, Parikh, fails to teach at least a shared process memory containing an inverted keyword index and an inverted keyword attribute index as recited in the Claims. Further, Parikh fails to teach generating an inverted keyword index and an inverted keyword attribute index corresponding to the keyword data. For these reasons, applicants respectfully request a withdrawal of the § 102(e) rejection with regard to Claims 1, 11, and 23.

b.   Dependent Claims 8, 20, and 30

Claims 8, 20, and 30 were rejected under 35 U.S.C. § 102(e) as being anticipated by Parikh. Because each dependent claim carries each and every limitation of the claim to which it

depends, Claims 8, 20, and 30 are allowable because of the previously presented arguments. Accordingly, Claim 8 depends on Claim 1, Claim 20 depends on Claim 11, and Claim 30 depends on Claim 24, which depends on Claim 23. In addition, Claims 8, 20, and 30 carry the further limitation of dynamically adjusting memory pointers corresponding to the inverted keyword index.

The section cited by the Office Action in Parikh is related to matching synonyms with keywords. This clearly does not teach adjusting the pointers in the shared memory to correspond to the address utilized by the query processing component. For this reason, applicants respectfully request a withdrawal of the § 102(e) rejection with regard to Claims 8, 20, and 30.

<div align="center">c.   <u>Dependent Claim 17</u></div>

Claim 17 was rejected under 35 U.S.C. § 102(e) as being anticipated by Parikh. Because each dependent claim carries each and every limitation of the claim to which it depends, Claim 17 is allowable because of the previously presented arguments. Accordingly, Claim 17 depends on Claim 16, which depends on Claim 11. In addition, Claim 17 carries the further limitation of inserting keyword attribute data corresponding to the keyword into a temporary keyword attribute index and converting the temporary keyword attribute index into the inverted keyword attribute index in the shared process memory buffer. Parikh teaches only the use of a final inverted keyword attribute index and stores that index into a file. Parikh, para. 0043, 0034-0036, 0043-0053, 0058-0065, 0086. For this reason, applicants respectfully request a withdrawal of the § 102(e) rejection with regard to Claim 17.

<div align="center">d.   <u>Dependent Claims 2-7, 9, 10, 12-16, 18, 19, 21, 24-29, and 31</u></div>

Claims 2-7, 9, and 10 are dependent on Claim 1. Claims 12-16, 18, 19, and 21 are dependent on Claim 11. Claims 24-29, and 31 are dependent on Claim 23. As discussed above, Parikh fails to teach or suggest all of the limitations recited with regard to Claims 1, 11, and 23.
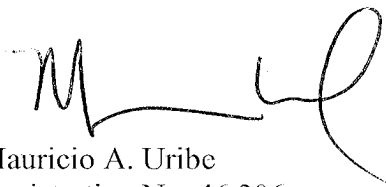
MSFT\22348AM.DOC

Accordingly, for the above-mentioned reasons, Claims 2-7, 9, 10, 12-16, 18, 19, 21, 24-29, and 31 are allowable over the cited art.

V. Conclusion

Based on the above-referenced arguments, applicants respectfully submit that all of the pending claims of the present application, Claims 1-21 and 23-31 are allowable over the cited and applied references. Accordingly applicants respectfully request withdrawal of all the rejections of the claims of the present invention and allowance of the present application. If any questions remain, applicants request that the Examiner contact the undersigned at the telephone number listed below.

Respectfully submitted,

CHRISTENSEN O'CONNOR
JOHNSON KINDNESS^{PLLC}

Mauricio A. Uribe
Registration No. 46,206
Direct Dial No. 206.695.1728

MAU:jmb/jlg

LAW OFFICES OF
CHRISTENSEN O'CONNOR JOHNSON KINDNESS^{PLLC}
1420 Fifth Avenue
Suite 2800
Seattle, Washington 98101
206 682 8100

-15-

MSF1\22348AM.DOC